File Implementation

The following access is required in order to perform file implementation functions:

- READ access to the special object type -I of the NSC external object type PRD-Docu-Object.
- If Implementation Plans are protected at object level, the appropriate access for a range of plans or fully qualified Plan ID (instances of the NSC external object type PRD-Docu-Object).

Additional Checks with File Implementation Functions

In addition the checks listed above, the following checks are performed depending on the file file Implementation function.

Execute implementation plan

- This function changes the status of a generation task, so the user needs at least MODIFY access to the Implementation Plan to be executed.
- Security checks are carried out for each generation task individually. The user needs the same access for the task as he would need for the corresponding generation function. See Generation.
- If a task cannot be executed due to security, the task is given the status sec. abort. The execution of the Implementation Plan continues.
- If an entire task cannot be executed due to security, the message "Implementation plan executed with errors" is given after generation.
- If the Implementation Plan is executed again, all tasks with the status sec. abort are automatically set to RE (reexecute). All other conditions must be set to RE manually.
  It is assumed that another user has different access rights or that the original user has acquired the necessary permission in the meantime.

Add / Extend implementation plan

- Only files to which the user has READ access can be entered in a plan (otherwise the user might be able to find out about the file via Restrictions, attributes or language).
- A user can add generation tasks for external members or external object types for which he does not have READ or MODIFY access, because no security-critical attributes are displayed with this function.
- If the Parameter with userviews is set to Y, generation tasks are first placed in the plan for the combination of File ID and external object types (depending on access rights).
  If the files specified under File ID contain a master file with userview(s), the userviews to which the user has READ access are also placed in the plan together with the corresponding external objects.

Extend implementation plan

- If a range of files is specified with asterisk notation, the files to which the user does not have READ access are not placed in the plan. No message is given to indicate how many files were suppressed due to security.
- If a range of files is specified with asterisk notation and the user is not authorized to access any files in is range, the message "Implementation Plan not extended" is given.

Modify / Display implementation plan

- If the user does not have READ access to the file, the file ID is displayed with the remark >>>protected<<<. The ID must be displayed, otherwise the plan’s contents would not be displayed completely. By displaying the ID it is sometimes possible to make a guess at the file type, but this information cannot be found out for certain.
- If the user does not have READ access to the file, generation functions are not displayed. This would make it possible to draw conclusions about certain attributes such as file type.
- The function codes that can be entered depend on the user’s access rights. The following access checks are performed.

Function Code	Check
DO,MO,OO	ADD or MODIFY access to the external object type or external object
DI, SM	Same check as with Administration function Display: READ access to external member.
IN,UN,RE	No additional checks.

Display Implementation Plan

- If the user has READ access to the file but no READ access to the generated member, the file ID and - in the following line - member and library name are displayed, the generation messages are suppressed with the remark >>>protected<<<.

